Puiseux展開プログラムの作成 (数式処理における理論と応用の研究)

元吉, 文男

数理解析研究所講究録
1199 巻 2001 年 173-178

Puiseux 展開プログラムの作成

産業技術総合研究所　元吉　文男(Fumio MOTOYOSHI) *

1　はじめに

現在 Java で作成中の数式処理プログラムにおいて、積分機能を充実させるために代数関数の不定積分を加えることを考えているが、そのためには、代数曲線の特異点での振る舞いを知る必要がある。そこで、代数曲線のすべての分岐を求めるプログラムの作成に着手し、基本的機能を実現したので報告する。

プログラムの機能としては以下のものとする。

- 入力: 代数曲線を表わす多項式 $G(x, y)$
- 出力: 全ての分岐 $\{F_i\}$

分岐は特異点が無限遠点になる場合でも同じ表現になるようにし、統一的に処理するようにしている。また計算は、近似数を使用せずに代数拡大体上で演算を行なうこととする。代数的数の表現に関しては、以前に発表したように、単一拡大による表現ではなく、なるべく簡単な拡大の組み合わせによる逐次拡大方法とし、それぞれの定義多項式の次数を小さくなるようにしている。

なお、概略の方針は次のとおりであり、次節以降で詳説する。

1. 全ての特異点を求める (含 無限遠点)。
2. 全ての分岐が求まるまで Puiseux 展開を行なう。

*moto@etl.go.jp

2 特異点の特定

代数曲線の特異点を求めるのに、アフィン座標のままで、

$$G = 0 \tag{1}$$
$$\frac{\partial G}{\partial x} = 0 \tag{2}$$
$$\frac{\partial G}{\partial y} = 0 \tag{3}$$

という連立方程式を解く方法があるが、代数曲線のすべての特異点を知る必要があるために、無限遠点も求めることになるので、

$$H(x,y,z) = z^n G(x/z, y/z) \tag{4}$$

として、射影座標に変換して次の連立方程式を解くことにした (n は G の全次数とする)。

$$\frac{\partial H}{\partial x} = 0 \tag{5}$$
$$\frac{\partial H}{\partial y} = 0 \tag{6}$$
$$\frac{\partial H}{\partial z} = 0 \tag{7}$$

この場合、$H=0$ の方程式は常に満たされるので不要となる。また、連立方程式をグレブナ基底を利用して解く場合には、同次式の方が解き易いことが知られており、効率的であると判断した。

ただし、Puiseux 展開の場合にはアフィン座標で考えており、次節で述べるように分岐を統一的に表現するために、特異点が無限遠点の場合に関しては以下の処理を行なうことにする。なお、以下ではアフィン座標で表現するときには (x,y) のように、射影座標で表現するときには $(x:y:z)$ のように記述する。

- (step 1)
  $(1:y_i:0)$ が特異点の場合には、アフィン座標で x 座標が無限大になるので、
  $$G(x,y) \leftarrow G(1/x, y) \tag{8}$$
  と変形し、新しい $G(x,y)$ について特異点を求め、次のステップに進む。また、変形したことを覚えておく。

- (step 2)
  $(0:1:0)$ が特異点の場合には、アフィン座標での有限の x が縮退して、すべてが 0 になってしまうため、$G(x,1/y)$ の特異点を上記の方法で求めて x 座標を求める。ただし、$G(x,y)$ はそのまま変形せずにおく。このとき、$G(x,1/y)$ の射影座標での特異点は $(x_i:0:1)$ であるので、$G(x,y)$ の特異点は (x_i,∞) と表現しておく。

3 各特異点での Puiseux 展開

前節で求めた特異点を $\{P_i = (x_i, y_i)\}$ とする。

まず、以降の処理の簡単化のために、各特異点 P_i について P_i を原点に平行移動する。

$$G(x, y) \quad \leftarrow \quad G(x - x_i, y - y_i) \tag{9}$$

ただし、y_i が ∞ のときは y_i については何もしない。

特異点がアフィン座標で無限遠点の場合にも統一的に分岐を表現できるように、以下のようにする。すなわち、特異点を (x_i, y_i) として

$$\begin{aligned} x - x_i &= t^{n_i}, \quad (10) \\ & \quad n_i\text{: integer} \\ y - y_i &= a_{i1}t^{m_{i1}} + a_{i2}t^{m_{i2}} + a_{i3}t^{m_{i3}} + \cdots, \quad (11) \\ & \quad m_{i1} < m_{i2} < m_{i3} < \cdots, \\ & \quad m_{ij}\text{: integer, } a_{ij} \neq 0 \end{aligned}$$

の組で分岐を表わす。

この表現では、

- x_i が無限大のときは $n_i < 0$ となる。
- y_i が無限大のときは $m_{i1} < 0$ となる。

ことになり、すべての特異点に関して同じ形で分岐が表現できている。

次に、$G(x, y)$ を原点に関して以下の手順で Puiseux 展開する。なお、この方法では、ステップ 2 の終了判定ができるように、原点を特異点とするすべての分岐について横型探索で展開を行なう。

Algorithm: Puiseux 展開

- (step 1) 初期化

$$G_1(t, y) \quad \leftarrow \quad G(t, y), \tag{12}$$
$$d \quad \leftarrow \quad 1, \tag{13}$$
$$k \quad \leftarrow \quad 1 \tag{14}$$

- (step 2) 終了判定
  d が前もって与えられた数より大きく、かつ原点を特異点に持つ全ての分岐での d の

合計が $G(0,y)=0$ の $y=0$ と $y=\infty$ における多重度の和と等しければ (step 7) へ行く。また、$G(t,y)$ が y を因子に持つときには、分岐の一つが正確に求まったことになり、それまでの級数を覚えておくが、他の可能性もあるので $G(t,y)$ から y の因子を取り除いたものについて計算を進める (ただし、この場合には (step 3) で最小次数が求まらないことがあり、そのときには終了する)。

- (step 3) 最小次数の決定

$$G_k(t,y) \quad = \quad \sum_{i,j} a_{ij} t^{u_i} y^{v_j}, \ \ (a_{ij} \neq 0) \tag{15}$$

としたときに、点の集合 $\{(v_i,u_j)\}$ のうち、その 2 点を結ぶ直線より下に他の点がこないような性質を持つ 2 点を選ぶ。その直線の傾きを g としたときに $g=-n_k/d_k$ を満たす互いに素な整数 n_k,d_k を求める。
$g<0$ であるのに、$y=\infty$ が特異点でないか $k>1$ の場合はこの 2 点は無視する。
一般的に、上記の性質を満たす傾きは複数個あり得るので、分岐の数がここで増える可能性がある。

- (step 4) 係数の決定

$$G_k(t^{d_k}, c_k t^{n_k}) \text{ の最小次数の係数} \quad = \quad 0 \tag{16}$$

を解いて c_k を求める。
ここでも、c_k は複数個の可能性があり得るので、分岐の数が増える可能性がある。

- (step 5) 次のステップの準備

$$\begin{aligned} G_{k+1}(t,y) &\leftarrow G_k(t^{d_k}, t^{n_k}(c_k+y)), & (17)\\ d &\leftarrow d\cdot d_k, & (18)\\ n_i &\leftarrow n_i\cdot d_k \quad (\text{for all } i<k), & (19)\\ k &\leftarrow k+1 & (20)\end{aligned}$$

- (step 6) 繰り返し
ステップ 2 へ行く。

- (step 7) 後処理
前の手順で特異点を求めるときに、$G(x,y)\leftarrow G(1/x,y)$ の変形をしていた場合には

$$d \quad \leftarrow \quad -d \tag{21}$$

とする。

各分岐について

$$x = t^d \quad (22)$$

$$y = t^{n_1}(c_1 + t^{n_2}(c_2 + \cdots)) \quad (23)$$

が求める Puiseux 展開になる。

以上のアルゴリズムでは、分数べきを扱わずに多項式の計算だけで処理を実現できるよう、(step 5) で $t \leftarrow t^{d_k}$ という置き換えを行なっている。また、後の計算での $G(t,y)$ 中の t のべき指数の大きさをおさえるために、通常では $y \leftarrow c_k t^{n_k} + y$ としているところを $t^{n_k}(c_k + y)$ という置き換えを行なっている。

4 おわりに

以上で、代数曲線のすべての分岐を求める方法を示したが、そこでは、まず、無限遠点を含むすべての特異点を射影座標を使用して求め、次に、求めた各点において Puiseux 展開を行なって分岐を求めている。また、この展開においては多項式計算だけで計算を実現できるようにアルゴリズムを作成している。

なお、この方法を実現するプログラムであるが、現状で完成しているのは以下のものである。

- 射影空間での特異点の求解
- $(0,0)$ と $(0,\infty)$ における Puiseux 展開

プログラムは Java で開発中の数式処理システム上に作成しており、既に作成されている代数拡大体での演算モジュールを利用している。

今後に残されているのは以下の部分である。

- 内部構造の変換 ($Q[x][y] \leftrightarrow Q[x,y]$)
  特異点を求める場合にはグレブナ基底の計算で $Q[x,y]$ の表現を利用しており、一方、Puiseux 展開を行なう場合には x の多項式を係数とする y の多項式という表現を使用しているために、相互の変換が必要である。
- アフィン空間と射影空間の変換
  アフィン座標で与えられた代数曲線を与える式から、特異点を求める場合には射影座標に変換し、Puiseux 展開する場合にはアフィン座標に変換する必要がある。

- 平行移動
  特異点を原点に移動するために必要。

これらの機能の実現は容易であり、手間だけの問題であるので早急に作成したい。